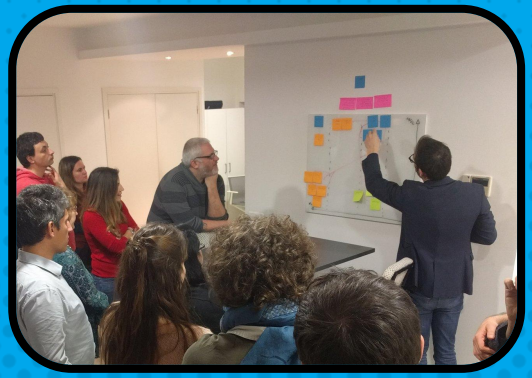
ANDROID ET IOS : ON S'EN FOUT, ON VOUS DIT TOUT.

Jordan Chapuy
Dev iOS
@Yupjoo
Rémy Barbosa
Dev Android
@BarbosaRmy

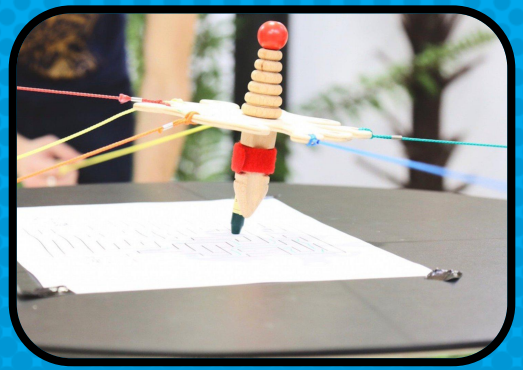
benext



Meetup

beCom

Events



Créativité

Fun

QUE FAIT-ON EN ENTREPRISE ?

ON S'AMUSE... PARFOIS (SOUVENT)

ON PRÉPARE AUSSI DES TALKS

BASICS

iOS

- ViewController
- Xib, Storyboard
- Obj-c / Swift
- XCode

Androïdeux

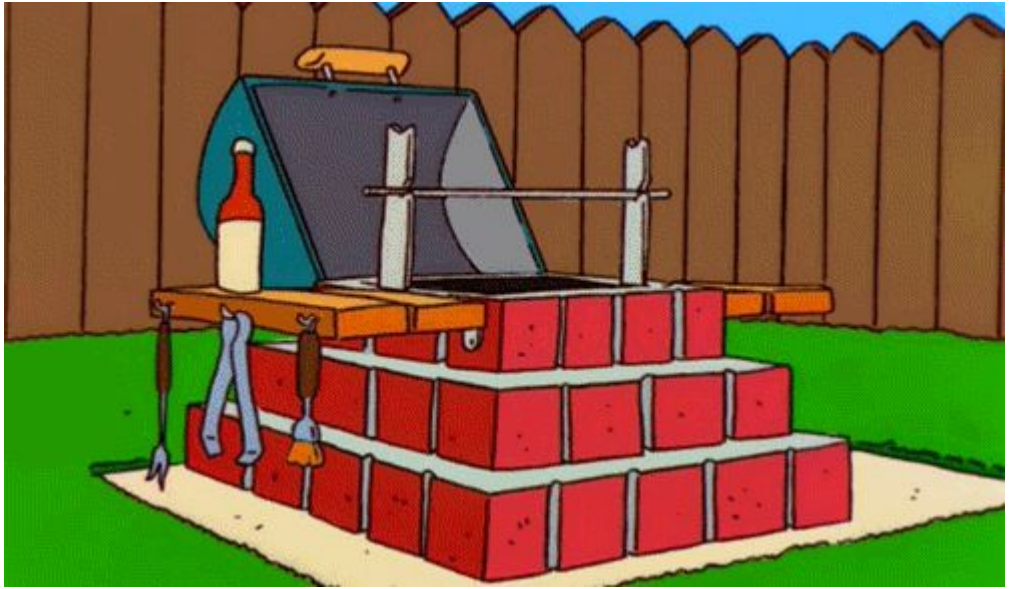
- Activity, Fragment
- xml, resource, etc.
- Java / Kotlin
- Android Studio

3.

RÉFLEXIONS ET ARCHITECTURE

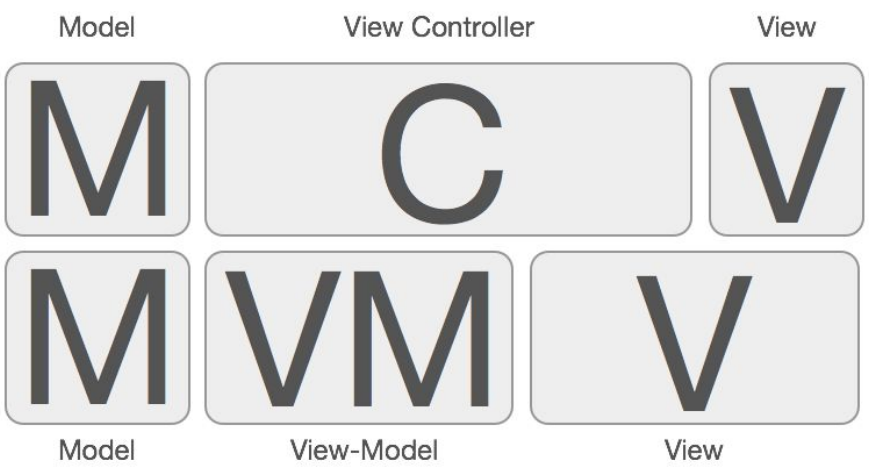
Avec ou sans plan IKEA ?

ARCHITECTURE

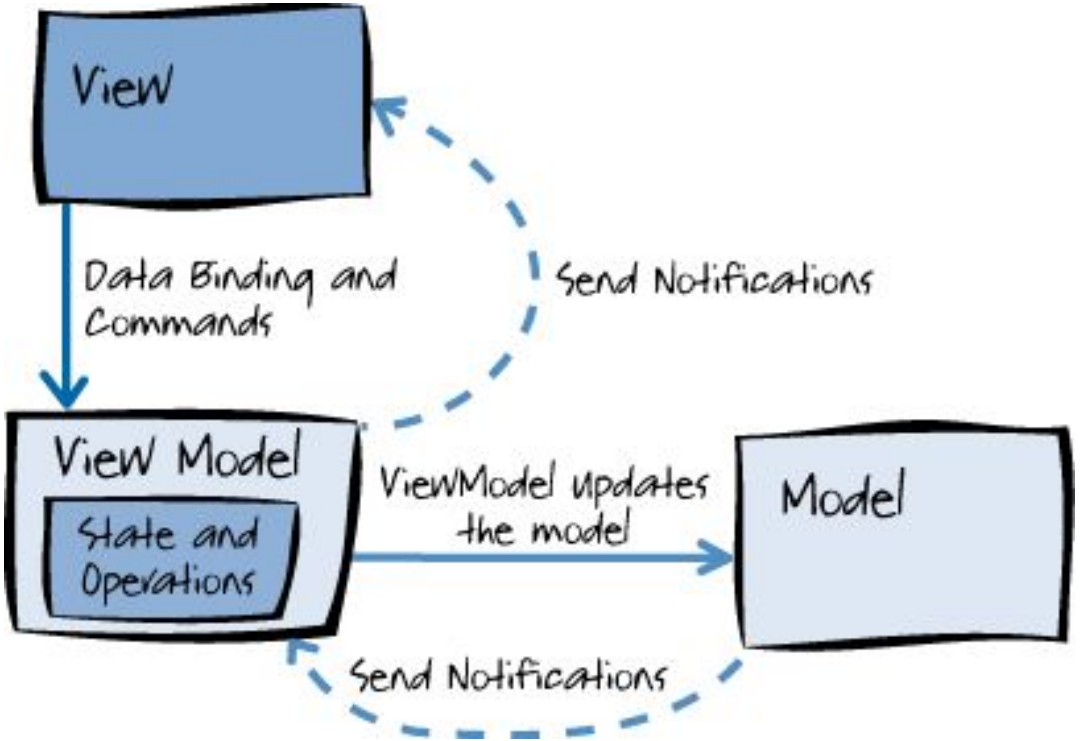


Architecture iOS : du MVC au MVVM

- Massive ViewController
- Meilleure séparation
- Responsabilités
- UIKit vs Tests Unitaires

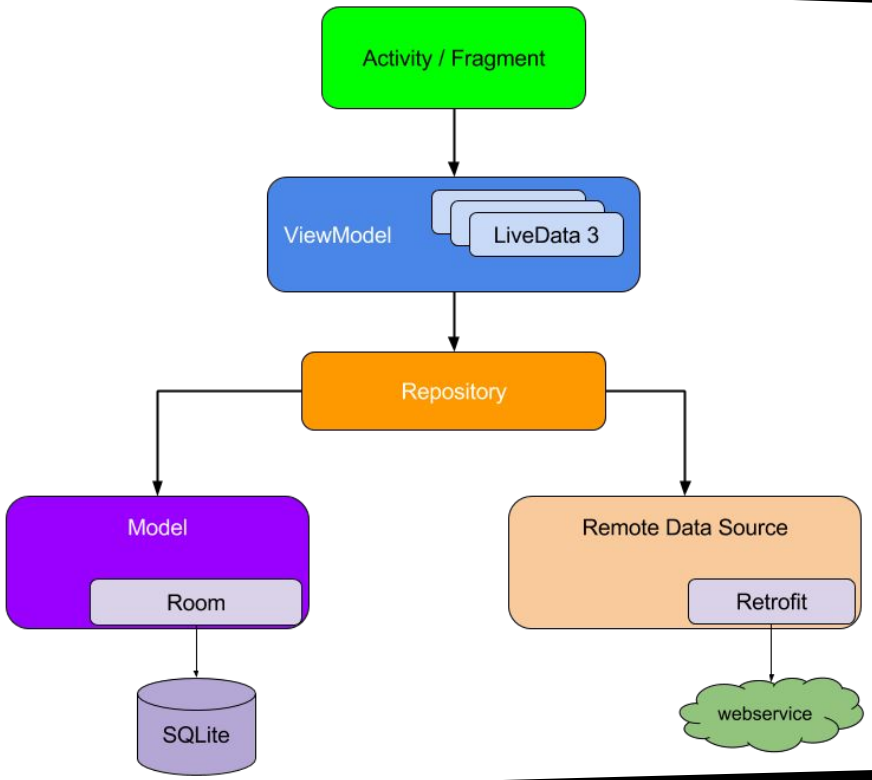


LE MVVM


View
Data Binding and Commands
Send Notifications
View Model
State and Operations
ViewModel updates the model
Model
Send Notifications

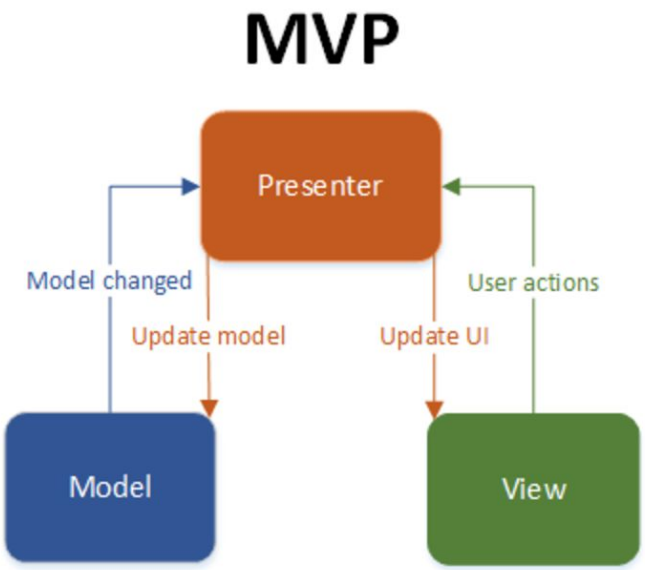
ARCHITECTURE ANDROID

- Vue
- Données à afficher
- Récupération des données



MVP

- Comparable au MVC ?
- La plus répandue sur Android
- Ecoute active du Presenter



ReactiveX

MET TOUT LE MONDE D'ACCORD

Explications de RX

- Basé sur le pattern Observer
- Unification callbacks et collections
- Une multitude d'opérateurs

Et porté sur plusieurs langages !

Petit exemple avec RxSwift

```
textFieldObservable

    .flatMap { text -> Observable<Data> in
        API.request(with: text)
    }

    .map { data -> [String] in
        return Parser.parse(data)
    }

    .bindTo(list)
```

```
textFieldObservable

    .observeOn(ConcurrentDispatchQueueScheduler(qos: .default))

3   .filter { $0.characters.count > 3 }

1   .debounce(0.200, scheduler: MainScheduler.instance)

2   .flatMapLatest { text -> Observable<Data> in
        API.request(with: text)
    }

    .map { data -> [String] in
        return Parser.parse(data)
    }

4   .observeOn(MainScheduler.instance)

    .bindTo(list)
```

3.

COOODEEEEEEER

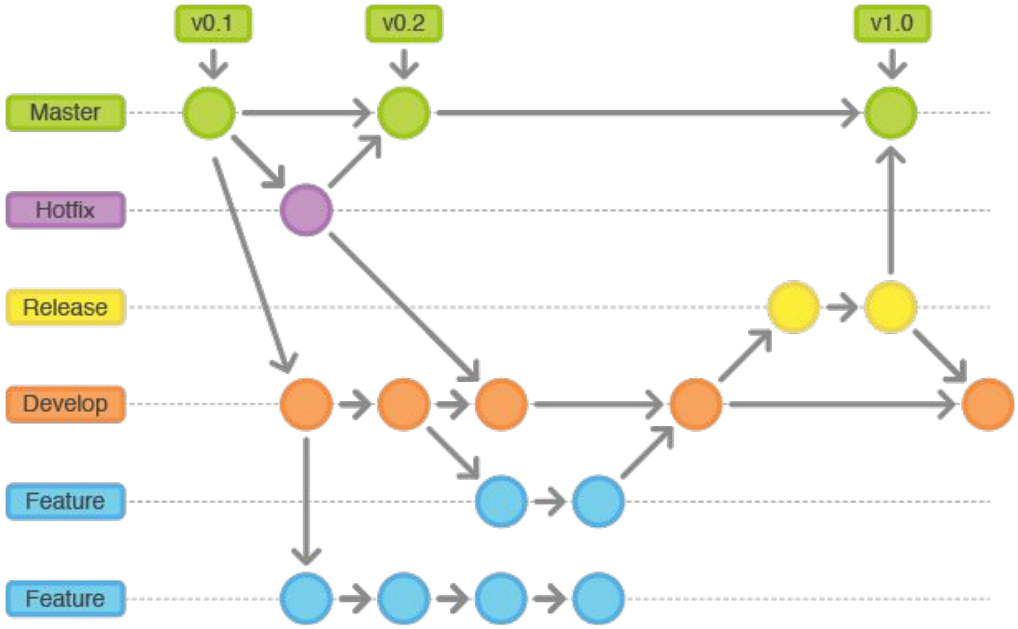
Turing n'a qu'à bien se tenir.

ON CODE COMME DES BOURRINS

ON OUBLIE PAS QUOI ?!

- Tests unitaires / TDD
- Tests UI
- Bonnes pratiques
- Conventions d'équipes

GIT / GITFLOW



MERGE REQUEST

- Revue de code
- Partage de connaissances

Request to merge feature/live-details_show-deta... into develop

Check out branch

Merge | Remove source branch | Squash commits | Modify commit message

You can merge this merge request manually using the command line

0 | 0

Discussion 3 | Commits 123 | Changes 279 | 0/2 discussions resolved

[redacted] added 3 commits about 3 hours ago

- 6a642456...65f2897e - 2 commits from branch develop
- f8bfa26f - Merge branch 'develop' into 'feature/live-details_show-details_easy-navigation-bar'

Compare with previous version

Jordan Chapuy @chapuy started a discussion on the diff about 2 hours ago

Last updated by Florent Capon about 2 hours ago

Toggle discussion

Echoes/Components/Coordinator/Coordinator.swift

```
8   8
9   9   import UIKit
10  10
11      - class Coordinator {
        11 + class Coordinator: NSObject {
```

Jordan Chapuy @chapuy commented about 2 hours ago

Owner

tu en as besoin pour quoi ?

3.

BOÎTE À OUTILS

Scie à béton et autres clés à pipe.

LES NERFS

Les images ont été réalisées par des professionnels.

Ne reproduisez pas ça chez vous.

Intégration continue

Jenkins

- Machine dédiée
- Build les commits
- Fait tourner les tests
- Produit les livrables

Et si on est bon ça balance sur le store !

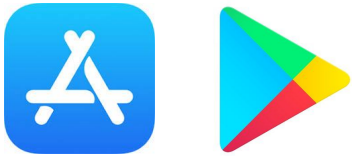
MÉTRIQUES

- Sonar
- Lint
- Qualité du code

Le tout dans un dashboard.

ON LIVREEEE !!!

- Store de beta
- Fastlane
- Crashlytics
- Continuous delivery



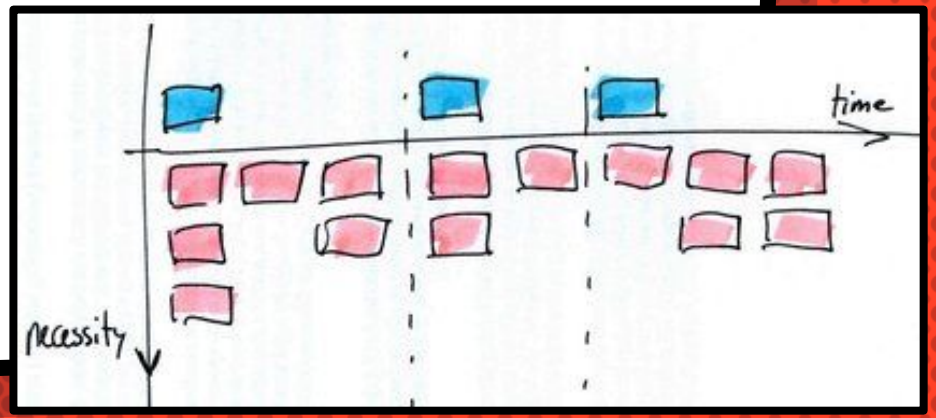
3.

CYCLE DE VIE PRODUIT AGILE

9 femmes font un bébé en 1 mois.
On peut tromper une fois 1000 personnes.
On aime les stats.

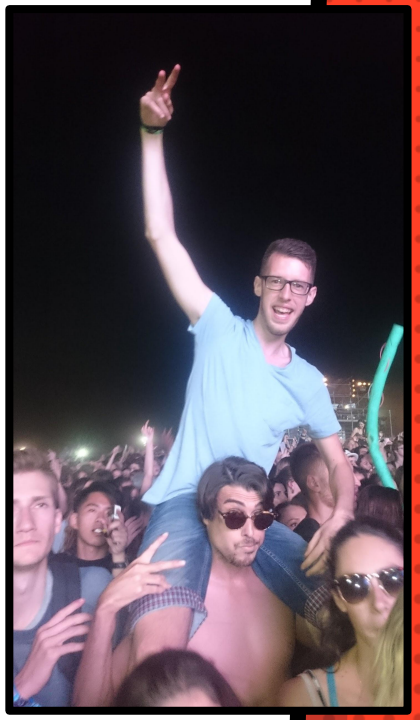
CONCEPTION

- Ateliers brainstorming
- User story mapping
- Ateliers utilisateurs
- Personas



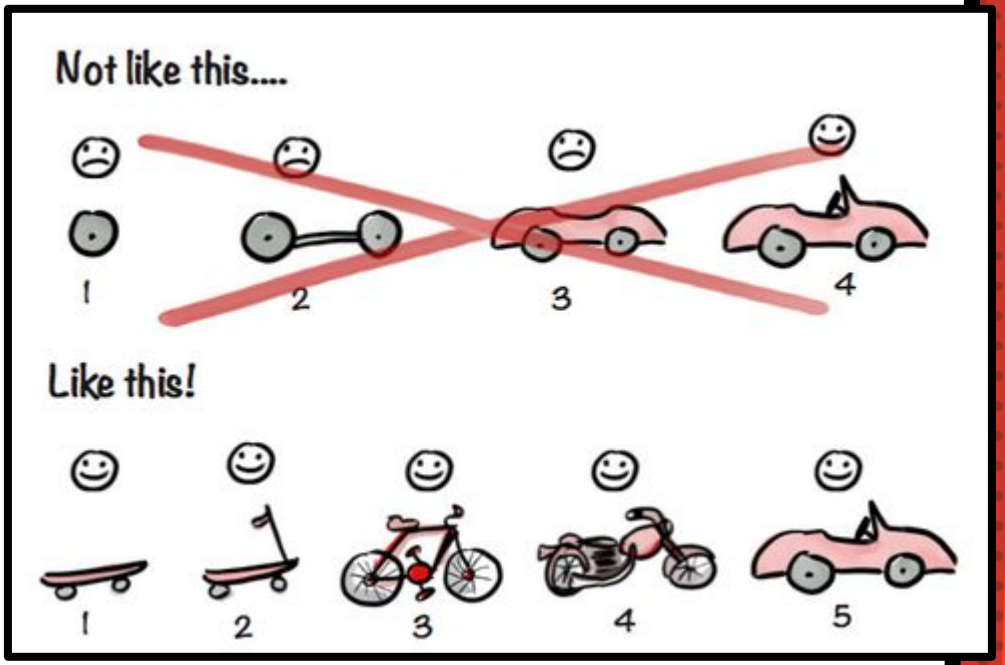
Création de l'équipe

- Équipe technique pluridisciplinaire
  - Devs
  - UI / UX
  - Product Owner
  - SCRUM master



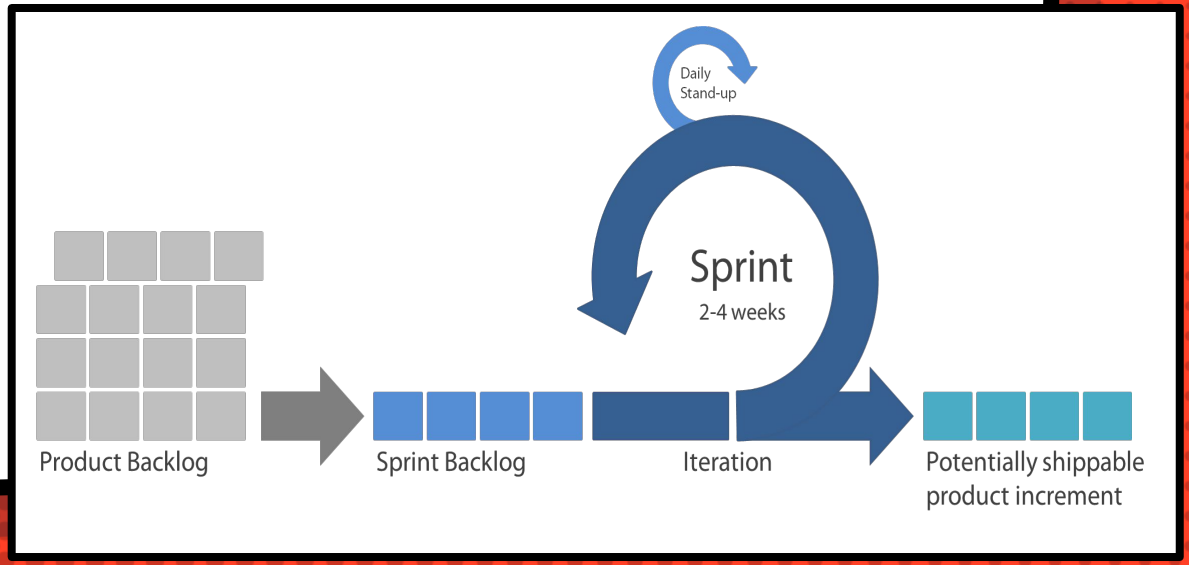
DÉFINITION MVP

- Minimum
- Valuable
- Product

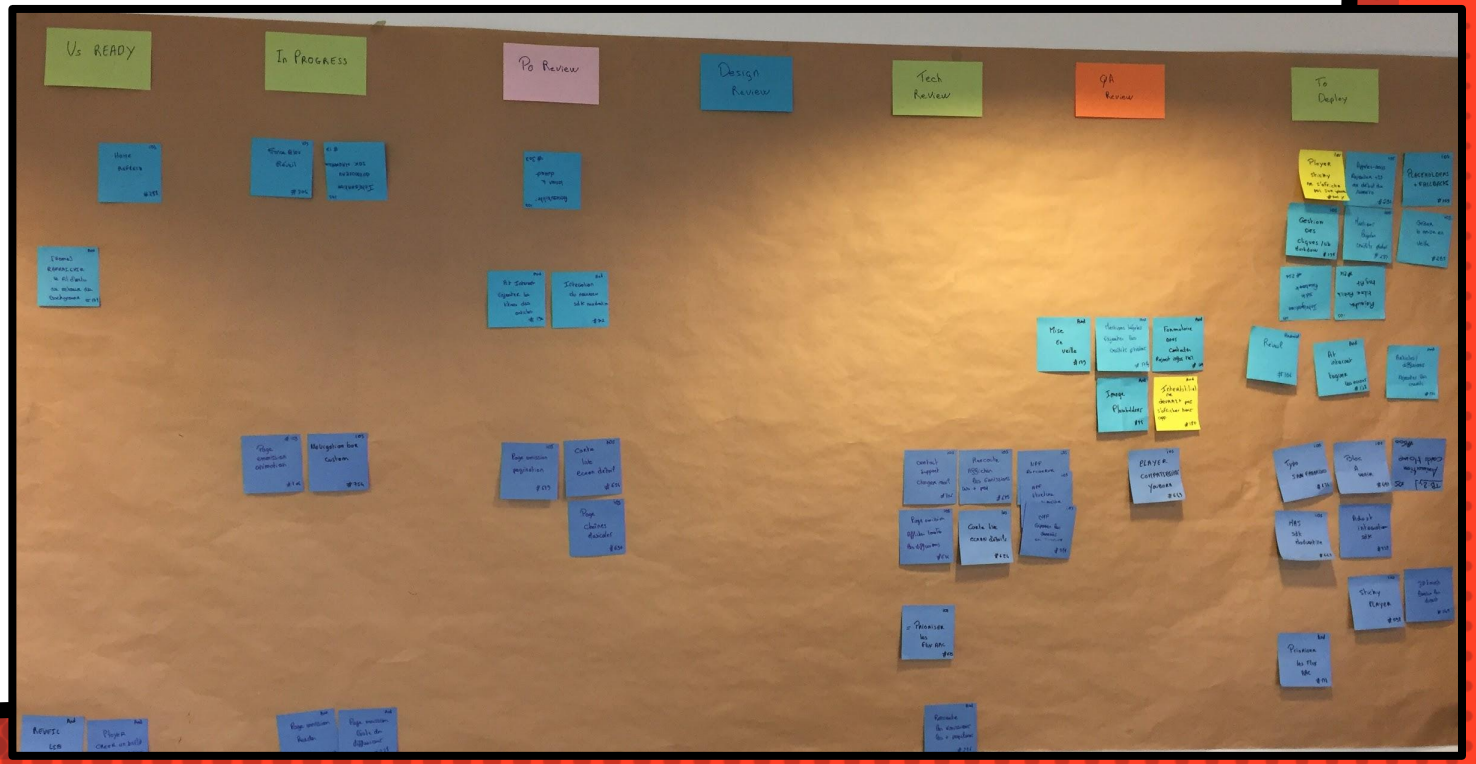


RÉALISATION

- SCRUM (un exemple parmi d'autres)



BOARD


Vs READY
In PROGRESS
Po Review
Design Review
Tech Review
QA Review
To Deploy

Réalisation

- Pour chaque user story
  - des tâches techniques
  - du dev
  - des binouzes 🍻

SOUMISSION

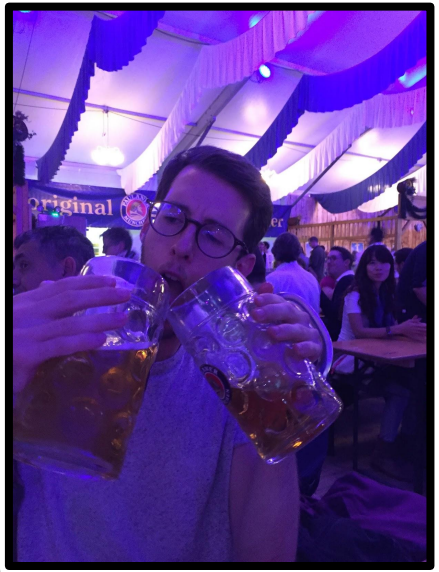
- On livre, et on suit le produit :
  - notes et commentaires
  - analytics
  - crashs

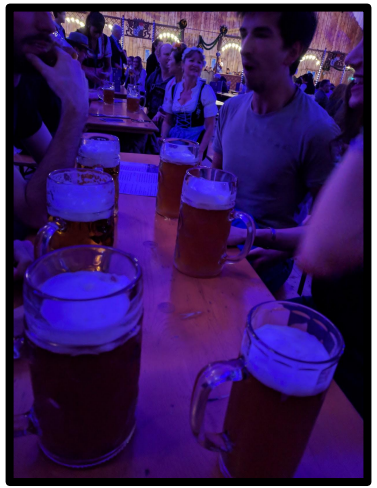
3.

ON RÉSUME ?

On prend les mêmes et on recommence.

LE PLUS IMPORTANT ? ... LA BIÈRE !





Réflexion en amont importante

- Archis plus poussés (MVVM / MVP)
- Concepts avancés (Rx)
- Testable (injection de dépendance, TDD)

On tends vers le craft !

Travail d'équipe

- Lint, convention de code
- Revue de code
- Process d'équipe
- Pair programming

Intégration continue

- Gitflow
- Jenkins
- Métriques
- Déploiement automatique

THANKS!

Retrouvez-nous sur Twitter @Yupjoo et @BarbosaRmy

Et au stand beNext dans l'école !